Stable Normal Forms for Polynomial System Solving

Bernard Mourrain
GALAAD, INRIA Méditerranée
2004 Route des Lucioles, BP 93,
06902 Sophia Antipolis, Cedex, France
Bernard.Mourrain@inria.fr

Philippe Trébuchet
UPMC,LIP6, Equipe APR
4 place jussieu
75015 Paris, France
Philippe.Trebuchet@lip6.fr

July 23, 2008

Abstract

This paper describes and analyzes a method for computing border bases of a zero-dimensional ideal I. The criterion used in the computation involves specific commutation polynomials and leads to an algorithm and an implementation extending the one provided in [29]. This general border basis algorithm weakens the monomial ordering requirement for Gröbner bases computations. It is up to date the most general setting for representing quotient algebras, embedding into a single formalism Gröbner bases, Macaulay bases and new representation that do not fit into the previous categories. With this formalism we show how the syzygies of the border basis are generated by commutation relations. We also show that our construction of normal form is stable under small perturbations of the ideal, if the number of solutions remains constant. This new feature for a symbolic algorithm has a huge impact on the practical efficiency as it is illustrated by the experiments on classical benchmark polynomial systems, at the end of the paper.

Keywords: Multivariate polynomial, quotient algebra, normal form, border basis, root-finding, symbolic-numeric computation.

1 Introduction

Solving polynomial systems is the cornerstone of many applications in domains such as robotics, geometric modeling, signal processing, chromatology, structural molecular biology etc. In these problems, the system has, most of the time, finitely many solutions and the equations often appear with approximate coefficients.

From a computational point of view, it is an actual challenge to develop efficient and stable methods to solve such problems. First backward stability is expected. The computed solutions should be the exact solutions of a system in the neighborhood of the input system. Efficiency is also mandatory to tackle the encountered polynomial systems. One can expect for instance that the behavior of the method depends mainly on the number of solutions, and partially on other extrinsic parameters such as the number of variables.

To handle the backward stability issue, one may consider classical numerical methods such as Newton-like iterations. However these local methods do not provide any guarantee of global convergence nor a complete description of all the roots. Algebraic methods, on the contrary, handle all the roots simultaneously. They reduce the problem to computing the structure of the quotient algebra $\mathcal{A}$ of the polynomial ring modulo the ideal I generated by the input system [8]. Such a structure is given by a basis B of $\mathcal{A}$ as a vector space, and the tables of multiplication in $\mathcal{A}$. Equivalently, it can be described by an algorithm of projection of the ring of polynomials $\mathbb{K}[\mathbf{x}]$ onto the vector space $\langle B \rangle$ generated by B, along the ideal I. We call such a projection, a normal form for the ideal I.

From the knowledge of the multiplication tables, we deduce either an exact encoding of the roots through a rational univariate representation [32], [12], or a numerical approximation by eigenvector computation, [1], [24], [34]. Since the eigencomputation can be considered as a numerically well-controlled process [13], the challenge becomes now to compute efficiently and in a stable way the quotient algebra structure $\mathcal{A}$.

In the family of algebraic methods, resultant-based techniques (see eg. [19], [9], [3]) exploit the properties of coefficients matrices of monomial multiples of the input equations in some specific degree. The table of multiplications are obtained by explicit Schur complements in these matrices [9]. Their construction is deeply linked to the geometry of the underlying variety that make these methods very tolerant against small perturbations. Unfortunately they heavily rely on genericity hypotheses that reduce their applicability. Moreover, the size of the constructed matrices usually grows exponentially with the number of variables.

To avoid such a pitfall, so-called H-bases have also been studied [20], [22]. They proceed degree by degree and stop when the terms of highest degree of the computed polynomials generate the terms of highest degree of all the ideal I. The stopping criterion requires the computation of generators of the syzygies of these highest degree terms. Though it also yields a basis of the quotient ring $\mathcal{A}$, without any apriori knowledge on its dimension, practically speaking, it also suffers from the swelling of the size of the linear systems to be solved.

The approach can be refined further by using a grading for which the highest term of a polynomial is a monomial. This leads to Gröbner basis computation (see eg. [6]). This approach also yields a basis of the quotient algebra $\mathcal{A}$ and a normal form for I. As in the other methods, their computation can be seen as just a triangulation of a certain matrix. Unfortunately these methods suffer from unavoidable instability: the monomial ordering attached to the Gröbner basis make the pivot selection strategy in the triangulation depend only on the symbolic structure of the rows of the matrix, and not on the numerical values of the coefficients appearing in these rows. This *can* lead to artificial unwanted singularities in the representation of the quotient ring (compare for instance the degree reverse lexicographic Gröbner basis of $p_1 = ax_1^2 + bx_2^2 + \varepsilon_1 x_1 x_2$, $p_2 = cx_1^2 + dx_2^2 + \varepsilon_2 x_1 x_2$ with $\varepsilon_1 = \varepsilon_2 = 0$ and $\varepsilon_1 \neq 0$, $\varepsilon_2 \neq 0$).

To circumvent this artificial difficulty, a new approach based on a normal form criterion which involves commutation relations (for bases B connected to 1) was first proposed in [25]. Further investigations of this technique [27], [28], including the Ph.D. dissertation [36] lead to a first version [29] of a normal form algorithm, which allows to construct efficiently and in a stable way zero-dimensional quotient algebra representations. Similar investigations were also pursued in [15], [16], but in the more restrictive case where the basis is stable by division. Other investigations related to the stabilization of the normal form process can also be found in [34].

In this paper, we discuss border basis algorithms in the general sense, that is for bases B which are connected to 1. See [26] for an introductory presentation of their properties. As in [11] or [10], our approach is based on linear algebra tools. As for H-bases, we use a grading of the polynomial ring, but the construction is optimized in the spirit of [27]. We describes an efficient criterion based on commutation polynomials to check the normal form property. This leads to an algorithm, as presented in [29], which has been improved to treat optimally the case when a syzygy of the components of highest degree is found, which is not a syzygy of the corresponding polynomials. We prove that the syzygies of a (general) border basis are generated by the commutation relations, giving a short and concise answer to a conjecture in [16] for basis stable by division. Meanwhile, works related to this conjecture for this special case were also investigated in [14]. Regarding the numerical stability of border bases, we prove that our construction of normal form is stable against small perturbations of the input system, if the number of solutions remains constant.

The paper is structured as follows. In the next section, we recall the notations, used in section 3 to prove the stopping criterion for generalized normal forms. In section 4, we show how to recover the syzygies from the commutation relations. In section 5, we recall briefly the maind idea of the algorithm described in [29]. In section 6, we analyze the stability of this algorithm from a symbolic-numeric perspective. Finally, we show the efficiency of our implementation and its numerical behavior on classical polynomial benchmarks.

2 Notations

We recall some of the definitions stated in [25], [27], [28] and add a few more that we will need in the sequel.

Let $\mathbb{K}$ be an effective field. The ring of n-variate polynomials over $\mathbb{K}$ will be denoted by R, $R = \mathbb{K}[\mathbf{x}] = \mathbb{K}[x_1, \ldots, x_n]$. We consider n-variate polynomials $f_1, \ldots, f_s \in R$. Our goal is to solve the system of equations $f_1 = 0, \ldots, f_s = 0$ over the algebraic closure $\overline{\mathbb{K}}$ of $\mathbb{K}$. These polynomials generate an ideal of $\mathbb{K}[\mathbf{x}]$ that we call I. The quotient of $\mathbb{K}[\mathbf{x}]$ modulo I will be denoted by $\mathcal{A}$. From now on, we suppose that *I is zero dimensional* so that $\mathcal{A}$ is a finite dimensional $\mathbb{K}$-vector space. The roots, with coordinates in the algebraic closure of $\mathbb{K}$, will be denoted by $\zeta_1, \ldots, \zeta_d$, with $\zeta_i = (\zeta_{i,1}, \ldots, \zeta_{i,n}) \in \overline{\mathbb{K}}^n$.

The support supp(p) of a polynomial $p \in \mathbb{K}[\mathbf{x}]$ is the set of monomials appearing in p with non-zero coefficients. Given a set S of elements of $\mathbb{K}[\mathbf{x}]$, we denote by $\langle S \rangle$ the $\mathbb{K}$-vector space spanned by the elements of S. Finally, we denote the set of all the monomials in the variables $\mathbf{x} = (x_1, \ldots, x_n)$ by $\mathcal{M}$. A term is an element of the form $\lambda \cdot m$ with $\lambda \in \mathbb{K} - \{0\}$ and $m \in \mathcal{M}$. For a subset S of $\mathcal{M}$, we will denote by S^c the set-theoretical complement of the set S in $\mathcal{M}$. For any monomial $m \in \mathcal{M}$, $m \cdot \mathcal{M}$ denotes the set of all monomial multiples of m.

For any subset S of R, we denote by S^+ or $D(S)$ the set $S^+ = S \cup x_1 S \cup \cdots \cup x_n S$, $\partial S = S^+ \backslash S$. S^+ is called the *prolongation* of S. For any $k \in \mathbb{N}$, $S^{[k]}$ is $S^{\overset{k \text{ times}}{+\cdots+}}$ the result of applying k times the operator $^+$ on S. By convention, $S^{[0]}$ is S.

If $B \subset \mathcal{M}$ contains 1, for any monomial $m \in \mathcal{M}$, there exists k such that $m \in B^{[k]}$. We say that a monomial m is of B-index k if $m \in B^{[k]} - B^{[k-1]}$, and we denote it by $\delta_B(m)$.

A set of monomials B is said to be connected to 1, if and only if, for every monomial m in B, there exists a finite sequence of variables $(x_{i_j})_{j\in[1,l]}$ such that $1 \in B$, $\forall l' \in [1, l]$, $\Pi_{j\in[1,l']} x_{i_j} \in B$ and $\Pi_{j\in[1,l]}\, x_{i_j} \ = m$.

A set of monomials B is stable by division, if for any $m \in B$ and any variable x_i such that $m = x_i\, m'$ ($m' \in \mathcal{M}$), we have m' in B. Remark that a set B stable by division, is connected to 1.

Definition 2.1 *Let Λ be a monoid with a well order relation $\prec$, such that:*

$$\forall \alpha, \beta, \gamma \in \Lambda,\ \ \alpha \prec \beta \Rightarrow \gamma + \alpha \prec \gamma + \beta$$

A $(\Lambda, \prec)$-grading of $\mathbb{K}[\mathbf{x}]$ is the decomposition of $\mathbb{K}[\mathbf{x}]$ as the direct sum: $\mathbb{K}[\mathbf{x}] = \bigoplus_{\lambda\in\Lambda} \mathbb{K}[\mathbf{x}]_{[\lambda]}$, with *the following property:*

$$\forall f \in \mathbb{K}[\mathbf{x}]_{[\alpha]},\ g \in \mathbb{K}[\mathbf{x}]_{[\beta]} \Rightarrow f\,g \in \mathbb{K}[\mathbf{x}]_{[\alpha+\beta]}.$$

We denote by degree of f or $\deg_\Lambda(f)$, or $\deg(f)$ (when no confusion is possible) and by $\Lambda(f)$, the following element of Λ:

$$\Lambda(f) = \min\{\lambda \in \Lambda \mid f \in \bigoplus_{\lambda' \preceq \lambda} \mathbb{K}[\mathbf{x}]_{[\lambda']}\}.$$

For any set $V \subset \mathbb{K}[\mathbf{x}]$, let $V_\lambda = \bigoplus_{\lambda' \preceq \lambda} \mathbb{K}[\mathbf{x}]_{[\lambda']} \cap V$. For any $\lambda \in \Lambda$, let $\lambda^+ = \min\{\lambda' \in \Lambda;\ \mathbb{K}[\mathbf{x}]_\lambda^+ \subset \mathbb{K}[\mathbf{x}]_{\lambda'}\}$ and let $\lambda^- = \max\{\lambda' \in \Lambda;\ \mathbb{K}[\mathbf{x}]_{\lambda'}^+ \subset \mathbb{K}[\mathbf{x}]_\lambda\}$.

In order not to be confused with different notions of degree, for any monomial m of $\mathbb{K}[\mathbf{x}]$, we define the size of m, denoted by $|m|$, to be the integer d such that $m = x_{i_1} \cdots x_{i_d}$, which itself imposes a grading.

Another classical grading is the one associated with a monomial ordering, where $\Lambda = \mathbb{N}^n$ and $\prec$ is a monomial order (see [7, p. 328]) such that for all $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{N}^n$, we have $\mathbb{K}[\mathbf{x}]_{[\alpha]} = \mathbb{K}\, x_1^{\alpha_1} \cdots x_n^{\alpha_n}$.

Definition 2.2 *We say that Λ is a reducing grading if Λ is a grading and if we have the property: for all monomials $m, m' \in \mathcal{M}$, such that m' divides strictly m, $\Lambda(m') \prec \Lambda(m)$, $\Lambda(m') \neq \Lambda(m)$.*

Both gradings induced by the classical degree and a monomial ordering are reducing grading. Hereafter, *we will denote by Λ a reducing grading.*

Definition 2.3 *A rewriting family F for a monomial set B is a set of polynomials $F = \{f_i\}_{i\in\mathcal{I}}$ such that* $\mathrm{supp}(f_i) \subset B^+$*, f_i has exactly* **one** *monomial $\gamma(f_i)$ (also called the leading monomial of f_i) in ∂B, if $\gamma(f_i) = \gamma(f_j)$ then $i = j$,*

Remark that the elements of F can be seen as rewriting rules for the leading monomial using monomials of B.

Definition 2.4 *A reducing family F of degree $\lambda \in \Lambda$ for a set B is a set of polynomials such that F is a rewriting family for B, $\forall m \in \partial B$ of degree at most λ, $\exists f \in F \mid \gamma(f) = m$.*

For the set $B = \{1, x_0, x_1, x_0x_1\}$, the set of polynomials $F = \{x_0^2 - 1, x_1^2 - x_1, x_0^2x_1 - x_1, x_1^2x_0 - x_1\}$ is a reducing family of degree 3.

Notice that a reducing family of degree λ for a set B (connected to 1) allows to rewrite the monomials of $\langle B^+\rangle_\lambda$ modulo F as elements of $\langle B\rangle_\lambda$. This leads in fact, to the definition of the linear projection π_F, associated to a reducing family for a set B connected to 1.

Definition 2.5 *Given a reducing family F of degree $\lambda \in \Lambda$ for a set B connected to 1, we define the linear projection $\pi_F : \langle B^+\rangle_\lambda \to \langle B\rangle_\lambda$ such that*

$$\begin{aligned}&\forall m \in B_\lambda, \pi_F(m) = m,\\ &\forall m \in \partial B_\lambda, \pi_F(m) = m - f;\end{aligned}$$

where $f \in F$ is the unique member of F such as $m = \gamma(f)$. We extend this construction to $\langle B^+\rangle_\lambda$ by $\mathbb{K}$-linearity.

There is a parallel with the differential algebra terminology, which we want to highlight here. To a polynomial $f(x_1,\ldots,x_n) \in R$ of degree at most δ, we can associate the differential equation $f(\partial_1,\ldots,\partial_n)\Phi(t_1,\ldots,t_n) = 0$, where ∂_i is the derivation with respect to the variable t_i. It is a linear equation in Φ. The solution $\Phi(t_1,\ldots,t_n)$ lives in the ring $\mathbb{K}[[t_1,\ldots,t_n]]$ of formal power series in $t_1,\ldots,t_n$, which we can truncate in degree $\lambda \succeq \Lambda(f_i)$. This set $\mathcal{J}_\lambda(\mathbf{t})$ of truncated series in degree λ is called the space of λ-*jets* in the literature. See eg. [33] for more details.

Given a set of polynomials $F \subset \mathbb{K}[\mathbf{x}]$ of degree at most λ, we consider the so-called *solution manifold* $\mathcal{R}_\lambda$ of the differential system $F(\partial)(\Phi) = 0$ in $\mathcal{J}_\lambda(\mathbf{t})$ (In our case, it is just a linear space). Given $\lambda' \prec \lambda$ and $\Phi \in \mathcal{J}_\lambda(\mathbf{t})$, we can forget the coefficient of degree bigger than λ' and project it on $\mathcal{J}_{\lambda'}(\mathbf{t})$. The set of solutions of $F(\partial)(\Phi) = 0$ projected on $\mathcal{J}_{\lambda'}(\mathbf{t})$, is then defined by the equations $F_{\lambda'}(\Phi') = 0$ where $\langle F_{\lambda'}\rangle = \langle F\rangle \cap \mathbb{K}[\mathbf{x}]_{\lambda'}$. We denote it by $\pi_{\lambda'}(F)$. We denote by $D(F)$ the new system which extends F with the $\partial_i f(\partial) = 0$, for $i \in 1\ldots n$, $f \in F$, obtained by formal multiplication by the ∂_i. It is called the prolongation of F.

The system F of degree at most λ is said *formally integrable* if for any $r \geq 0$, $\pi_\lambda(D^{r+1}(F)) = F$. A technical condition of involutivity is introduced to ensure the regularity of the differential system [30]. A result of Cartan-Kuranishi [4, 17, 30] asserts that any system F can be transformed by prolongation and projection into a (involutive) formally integrable system. The connection of this construction with the notion of Mumford regularity of polynomial systems has been detailed in [21]. This correspondence has been used explicitly for solving polynomial equations in [31]. This involutive division is also used to construct a family of normal form projection related to so-called involutive bases (see eg. [2]). In Janet basis construction, however one difference is that the variables do not play a symmetric role. So-called multiplicative variables are used to perform reduction and non-multiplicative to extend the polynomial vector space.

In the following, instead of working in the vector space of all polynomials of a given degree, we will consider completion procedures based on prolongations and projections relative to a specific set of monomials B. Dealing locally with this set of monomials related to the number of solutions of the system will allow us to improve significantly the linear algebra stages in this type of methods.

In the sequel, we will make a heavy use of multiplication operators by one variable that we define as follows:

$$\begin{array}{rcl} M_{i,\lambda} : \langle B\rangle_{\lambda^-} & \to & \langle B\rangle_\lambda \\ b & \mapsto & \pi_F(x_i b). \end{array}$$

The subscript $_\lambda$ is redundant as soon as we know that F is a reducing family of degree λ, and we will omit this subscript in the sequel.

Definition 2.6 *Let $F = \{f_1,\ldots,f_s\}$ be a polynomial set, we denote by $F_{\langle\lambda\rangle}$ the vector space:*

$$F_{\langle\lambda\rangle} = \langle\{x^\alpha f_i |\ \Lambda(x^\alpha f_i) \leq \lambda\}\rangle.$$

Obviously, we have $F_{\langle\lambda\rangle} \subset (F)_\lambda$ where (F) is the ideal generated by F. Next we introduce a definition, which is weakening the notion of monomial ordering for Gröbner basis:

Definition 2.7 *We say that a function $\gamma : \mathbb{K}[\mathbf{x}] \to \mathcal{M}$ ($\mathcal{M}$ is the set of all monomials in the variables $x_1,\ldots,x_n$), is a choice function refining the grading Λ, if for any polynomial p, $\gamma(p)$ is a monomial such that $\gamma(p) \in \mathrm{supp}(p)$, if $m \in \mathrm{supp}(p)$, $m \neq \gamma(p)$ then $\gamma(p)$ does not divide m, and $\Lambda(\gamma(p)) = \max\{\Lambda(m),\ m \in \mathrm{supp}(p)\}$. The coefficient of the monomial $\gamma(p)$ in p will be denoted by $\kappa(p)$.*

Example 2.8 *In the following, we consider a* Macaulay[1] *choice function γ, such that for all $p \in \mathbb{K}[\mathbf{x}]$, $\gamma(p) = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$ satisfies, $|\gamma(p)| = \max\{|m|; m \in \operatorname{supp}(p)\} = d$, and $\exists i_0$ st. $\alpha_{i_0} = \max\{\deg_{x_i}(m), m \in \operatorname{supp}(p)$ and $, |m| = d; i = 1, \ldots, n\}$. In case more than one monomial satisfy these conditions, the greatest monomial for the lexicographic order is chosen.*

The monomial returned by the choice function has the same name as the *leading* monomial of an element of a reducing family. This is intended, as we will define a reducing family on the behalf of choice functions, and in this framework the two will coincide. Hereafter if $S = \{p_1, \ldots, p_s\}$ is a polynomial set, then we denote by $\gamma(S)$ the set: $\gamma(S) = \{\gamma(p_1) \ldots \gamma(p_s)\}$.

Definition 2.9 *Let γ be a choice function refining a grading Λ. For any polynomials $p_1, p_2 \in \mathbb{K}[\mathbf{x}]$, let the C-polynomial relative to γ and (p_1, p_2) be*

$$C(p_1, p_2) = \frac{\operatorname{lcm}(\gamma(p_1), \gamma(p_2))}{\kappa(p_1)\,\gamma(p_1)} p_1 - \frac{\operatorname{lcm}(\gamma(p_1), \gamma(p_2))}{\kappa(p_2)\,\gamma(p_2)} p_2.$$

Let the C-degree of (p_1, p_2) be $\Lambda(\operatorname{lcm}(\gamma(p_1),\ \gamma(p_2)))$ and let the leading monomial of the pair (p_1, p_2) be $\operatorname{lcm}(\gamma(p_1), \gamma(p_2))$.

This is almost the same definition as a S-polynomial [6] when γ is a monomial ordering. We however use a new name to underline that now γ may not be a monomial ordering (i.e. a total order compatible with monomial multiplication). As we will see in the next section, the C-polynomials express commutation conditions for the $M_{i,\lambda}$.

3 Generalized normal form criterion

Let $F = \{f_1, \ldots, f_s\}$ be a polynomial system and let I be the ideal generated by F. Remember that $F_{\langle\lambda\rangle}$ (the $\mathbb{K}$-vector space spanned by the monomial multiples of the f_i, $x^\alpha f_i$ of degree $\preceq \lambda \in \Lambda$) is included in I_λ. Thus, when $I_\lambda = F_{\langle\lambda\rangle}$ we can define a normal form modulo I, up to the degree λ as the projection of $\mathbb{K}[\mathbf{x}]_\lambda$ along $F_{\langle\lambda\rangle}$ onto a supplementary space $\langle B\rangle_\lambda$. Hereafter, we consider a set B of monomials, containing 1.

Let F be a rewriting family, and let $\mathcal{H} = \{m,\ \exists p \in F, \gamma(p) = m\}$ be the set of their leading monomials then, obviously F allows us to define the projection π_F of $B \cup \mathcal{H}$ on B along $\langle F\rangle$. However we may extend this projection using the following extension process:

Definition 3.1 *Let F be a rewriting family. For all $m \in B$, we define $\pi_F^e(m) = m$. For $m \notin B$, there exists $m' \in \partial B$ and r integers $i_1, \ldots, i_r \in [1, n]$, such that $m = x_{i_1} \cdots x_{i_r} m'$. We define $\pi_F^e(m)$ by induction on k, as follows.*
- *if $r = 0$, $\pi_F^e(m')$ is defined as $\pi_F^e(m') = \pi_F(m') = m' - f$ where $f \in F$ is such that $\gamma(f) = m'$.*
- *$\forall k, 1 \leq k \leq r$, $\pi_F^e(x_{i_{r-k}} \cdots x_{i_r} m') = \pi_F(x_{i_{r-k}} \pi_F^e(x_{i_{r-k+1}} \cdots x_{i_r} m'))$, if this latter quantity is defined.*

Otherwise we say that $\pi_F^e(m)$ is undefined.

Remark that the above process allows us to define π_F^e only on monomials, and we extend it implicitly, by linearity. Remark also that this extension process is not defined in a unique way. Indeed, two different decompositions of a monomial m may lead to two different values of $\pi_F^e(m)$. However the following theorem shows that this extension process becomes canonical as soon as we check some commutativity conditions.

[1] It gives monomial basis similar to those given by Macaulay in his multivariate resultant construction [19].

Theorem 3.2 *Assume that B is connected to 1. Let F be a rewriting family, and let E be the set of monomials m such that for all decomposition of m as a product of variables, $m = x_{i_0} \cdots x_{i_k}$, $\pi_F(x_{i_0}\pi_F(x_{i_1}\cdots\pi_F(x_{i_k})\cdots))$ is defined. Suppose that for all $m \in E$ and all indexes $i, j \in [1, n]$ such that $x_i x_j m \in E$, we have:*

$$\pi_F^e(x_i \pi_F^e(x_j m)) = \pi_F^e(x_j \pi_F^e(x_i m)).$$

Then π_F^e coincides with the linear projection π_S of $\langle E\rangle$ on $\langle B\rangle$ along the vector space spanned by the polynomials $S = \{x^\alpha f,\ \alpha \in \mathbb{N}^n,\ f \in F \text{ and } x^\alpha\gamma(f) \in E\}$.

Proof. Remark that the way we define it, makes π_F^e inherently a *linear* multivalued map. Hence to prove the theorem we have first to show that under the above hypotheses, π_F^e becomes a well defined map, and next that this well defined linear map coincide with the projection π_S of $\langle E\rangle$ on $\langle B\rangle$ along $\langle S\rangle$.

Remark also that E is obviously stable by monomial division: if all the possible decompositions of m as a product of variables $m = x_{i_0} \cdots x_{i_k}$ are such that $\pi_F(x_{i_0} \cdots \pi_F(x_{i_k}))$ is defined, then a fortiori if m' is a divisor of m, this property is true for m'. Let us show that the extension process is independent of the way m is decomposed as a product of variables. Let $m = x_{i_0}\, m' = x_{i_1}\, m''$ with $i_0 \neq i_1$ and $m, m', m'' \in E$, then there exists $m''' \in E$ (since E is stable by monomial division) such that $m = x_{i_0}\, x_{i_1} m'''$. As $m,\ m',\ m''$, and m''' are in E, $\pi_F^e(m')$, $\pi_F^e(m'')$, $\pi_F^e(x_{i_0}m')$, $\pi_F^e(x_{i_1}m'')$, and $\pi_F^e(m''')$ are defined and we have:

$$\begin{array}{rcl}\pi_F^e(x_{i_0}\,\pi_F^e(m')) & = & \pi_F^e(x_{i_0}\,\pi_F^e(x_{i_1}\pi_F^e(m'''))),\\ \pi_F^e(x_{i_1}\,\pi_F^e(m'')) & = & \pi_F^e(x_{i_1}\,\pi_F^e(x_{i_0}\pi_F^e(m'''))).\end{array}$$

The commutation condition guarantees that the two quantities are equal, so that the definition of π_F^e does not depend on the way to write m as a product of variables.
Next we have to show that π_F^e and π_S coincide on their common set of definition. We do it by induction on the size of the monomials:
It is true that $\pi_F^e(1) = \pi_S(1) = 1$ (since $1 \in B$). For any monomial $m \neq 1$ in E, the property of connectivity of B and the definition of E gives us: $\exists m' \in E$ and $i_0 \in [1, n]$ such that $m = x_{i_0}m'$ and $\pi_F^e(m')$ is defined, so that we have:

$$\pi_F^e(m) \quad = \quad \pi_F^e(x_{i_0}m') =^{def} \pi_F^e(x_{i_0}\pi_F^e(m')) =^{induction} \pi_F^e(x_{i_0}\pi_S(m')) \in \langle B\rangle.$$

Now by induction, $m' - \pi_S(m') \in S_{\lambda^-}$ where $\lambda = \Lambda(m)$ and

$$m - \pi_F^e(m) = x_{i_0}(m' - \pi_S(m')) + (x_{i_0}\pi_S(m') - \pi_F^e(x_{i_0}\pi_S(m'))) \in \langle S\rangle.$$

Thus $\pi_F^e(m)$ is the projection of m on $\langle B\rangle$ along $\langle S\rangle$. □

Suppose now that we are given a reducing family of degree δ instead of a rewriting family. Then we can further extend the above theorem with the help of the following lemma.

Lemma 3.3 *Let F be a reducing family of degree λ for a set B connected to 1, and suppose that $\forall f \in F,\ \Lambda(\gamma(f)) = \Lambda(f)$. With the notation of Theorem 3.2, the set E contains the set of monomials of degree less than or equal to λ.*

Proof. Let $m \in \mathcal{M}_\lambda$ be a monomial of degree less or equal to λ, then m can be written as $m = x_{i_1} \ldots x_{i_d}$ with $d = |m|$. Let us prove by induction on $k \leq d$, that $p_k = \pi_F(x_{i_k}\pi_F(\cdots\pi_F(x_{i_1}))\cdots)$ is defined and that $\Lambda(p_k) \leq \Lambda(x_{i_k}\cdots x_{i_1})$.

Consider now $x_{i_{k+1}} p_k \in B^+$. For $m' \in \mathrm{supp}(x_{i_{k+1}} p_k) \cap B$, we have $\pi(m') = m'$ and by induction hypothesis $\Lambda(m') \leq \Lambda(x_{i_{k+1}} \cdots x_{i_1})$. For $m' \in \mathrm{supp}(x_{i_{k+1}} p_k) \cap \partial B$, as F is a reducing family of degree λ, we have a rewriting rule for m'. The hypothesis that $\forall f \in F,\ \Lambda(\gamma(f)) = \Lambda(f)$ implies that m' rewrites in terms of monomials of degree bounded by $\Lambda(x_{i_{k+1}} \cdots x_{i_1})$. This proves $p_{k+1} := \pi_F(x_{i_{k+1}} p_k)$ is defined and $\Lambda(p_{k+1}) \leq \Lambda(x_{i_{k+1}} \cdots x_{i_1})$.
This proves by induction that $\pi_F^e(x_{i_1} \cdots \pi_F^e(x_{i_d}))$ is defined, for any decomposition $m = x_{i_1} \cdots x_{i_d} \in \mathcal{M}_\lambda$ so that $m \in E$. This ends the proof. □

Theorem 3.4 *Let F be a reducing family of degree λ for a set B connected to 1. If we have:*

- $\forall f \in F,\ \Lambda(\gamma(f)) = \Lambda(f)$.
- $M_{j,\lambda} \circ M_{i,\lambda^-} = M_{i,\lambda} \circ M_{j,\lambda^-}$*, for* $1 \leq i, j \leq n$,

then, we can extend π_F to a linear projection π_F^e from $\mathbb{K}[\mathbf{x}]_\lambda$ onto $\langle B \rangle_\lambda$ with kernel $F_{\langle \lambda \rangle}$.

Proof. As F is a reducing family of degree λ, by Lemma 3.3, we have $E \supset \mathcal{M}_\lambda$.

Let us prove that for all $m \in \mathcal{M}_{\lambda^{--}}$ and all pairs of indices (i, j), there exists a way to define π_F^e such that $\pi_F^e(x_i\ \pi_F^e(x_j m)) = \pi_F^e(x_j\, \pi_F^e(x_i m))$.

As $\mathcal{M}_{\lambda^{--}} \subset \mathcal{M}_\lambda \subset E$, $\pi_F^e(m)$ is defined and $\mathrm{supp}(\pi_F^e(m)) \subset B$. We define $\pi_F^e(x_i m) = \pi_F(x_i \pi_F^e(m))$ and, similarly, $\pi_F^e(x_j m) = \pi_F(x_j \pi_F^e(m))$. With this definition we have:

$$\begin{aligned} \pi_F^e(x_i \pi_F^e(x_j m)) &= M_{i,\lambda}(M_{j,\lambda^-}(\pi_F^e(m))) \\ &= M_{j,\lambda}(M_{i,\lambda^-}(\pi_F^e(m))) = \pi_F^e(x_j \pi_F^e(x_i m)). \end{aligned}$$

which proves the commutation property. We end the proof by applying Theorem 3.2. □

Corrolary 3.5 *With the hypothesis of Theorem 3.4, we have $\mathbb{K}[\mathbf{x}]_\lambda = \langle B \rangle_\lambda \oplus F_{\langle \lambda \rangle}$.*

Let us give here another effective way to check that we have a projection from $F_{\langle \lambda \rangle}$ (vector space spanned by the monomial multiples of the f_i of degree λ) onto $\langle B \rangle_\lambda$ (element of degree λ of the vector space spanned by B) starting from a reducing family of degree λ, without computing *explicitly* the multiplication operators.

Theorem 3.6 *Let $\lambda \in \Lambda$. Let F be a reducing family of degree $\lambda \in \Lambda$, for B. Assume that $\forall\ f \in F,\ \Lambda(\gamma(f)) = \Lambda(f)$ and let π_F be the induced projection from $\langle B^+ \rangle_\lambda$ onto $\langle B \rangle_\lambda$. Then $\forall f, f' \in F_{\langle \lambda \rangle}$ such that $C(f, f') \in \langle B^+ \rangle_\lambda$,*

$$\pi_F(C(f, f')) = 0$$

iff π_F extends uniquely as a projection π_F^e from $\mathbb{K}[\mathbf{x}]_\lambda$ onto $\langle B \rangle_\lambda$ such that $\ker(\pi_F^e) = F_{\langle \lambda \rangle}$.

Proof. By Theorem 3.4, we have to show that this condition is equivalent to the commutation of the operators $M_{i,\lambda'}, \lambda' < \lambda$ on the monomials of $B_{\lambda^{--}}$.
For any $m \in B_{\lambda^{--}}$ and any $i_1 \neq i_2$ such that $x_{i_1} m \in \partial B$, $x_{i_2} m \in \partial B$, there exists $f, f' \in F_{\langle \lambda^- \rangle}$ such that $\gamma(f) = x_{i_1} m$, $\gamma(f') = x_{i_2} m$. Thus, we have $\pi_F(x_{i_1} m) = \gamma(f) - f$, $\pi_F(x_{i_2} m) = \gamma(f') - f'$ and $C(f, f') = x_{i_2} f - x_{i_1} f' \in \langle B \rangle_\lambda^+$. Consequently,

$$\begin{aligned} & M_{i_2,\lambda}(M_{i_1,\lambda^-}(m)) - M_{i_1,\lambda}(M_{i_2,\lambda^-}(m)) \\ &= M_{i_2,\lambda}(\gamma(f) - f) - M_{i_1,\lambda}(\gamma(f') - f') \\ &= \pi_F(x_{i_2}\gamma(f) - x_{i_2} f) - \pi_F(x_{i_1}\gamma(f') - x_{i_1} f') \\ &= \pi_F(x_{i_1} f' - x_{i_2} f) = \pi_F(C(f', f)). \end{aligned}$$

which is zero by hypothesis. A similar proof applies if $x_{i_1}\, m \in B$ or $x_{i_2}\, m \in B$.

Conversely, since $\ker(\pi^e_F) = F_{\langle\lambda\rangle}$ and $C(f, f') \in F_{\langle\lambda\rangle} \cap \langle B\rangle^+_\lambda$, we have that $\pi_F(C(f', f)) = \pi^e_F(C(f', f)) = 0$, which proves the equivalence and Theorem 3.6. □

Remark 3.7 *In this proof, we have shown that if the C-polynomials up to the degree λ reduce to 0, then the multiplication operators $M_{i,\lambda}$ commute.*

Definition 3.8 *A reducing family F for all degrees $\lambda \in \Lambda$ on a set B of monomials, connected to 1 will be called a* border basis *for B.*

Finally, the previous results leads to a new proof of Theorem 3.1 of [25]:

Theorem 3.9 *Let F be a border basis for a set B of monomials, connected to 1, let π_F be the corresponding reduction from $\langle B^+\rangle$ onto $\langle B\rangle$, and let $M_i : \langle B\rangle \to \langle B\rangle$ such that $\forall\, b \in \langle B\rangle$, $M_i(b) = \pi_F(x_i\, b)$. Then,*

$$M_j \circ M_i = M_i \circ M_j, \text{ for } 1 \leq i, j \leq n$$

iff there exists a unique projection π^e_F from $\mathbb{K}[\mathbf{x}]$ onto $\langle B\rangle$ such that $\ker(\pi^e_F) = (F)$ and $(\pi^e_F)_{|\langle B^+\rangle} = \pi_F$.

Proof. Under these hypotheses, by Theorem 3.4, for any $\lambda \in \Lambda$, $(\pi_F)_{|\langle B^+\rangle_\lambda}$ extends uniquely to a projection $\pi^e_{F_\lambda}$ from $\mathbb{K}[\mathbf{x}]_\lambda$ onto $\langle B\rangle_\lambda$, such that $\ker(\pi^e_{F_\lambda}) = F_{\langle\lambda\rangle}$. Since for any $\lambda, \lambda' \in \Lambda$ such that $\lambda \prec \lambda'$, we have $(B_{\lambda'})_\lambda = B_\lambda$, and $F_{\langle\lambda\rangle} \subset (F_{\langle\lambda'\rangle})_\lambda$, we also have $(\pi^e_{F_{\lambda'}})_{|\mathbb{K}[\mathbf{x}]_\lambda} = \pi^e_{F_\lambda}$. This defines a unique linear operator π^e_F on $\mathbb{K}[\mathbf{x}]$ such that $\pi^e_{|\mathbb{K}[\mathbf{x}]_\lambda} = \pi^e_{F_\lambda}$ and $\ker(\pi^e_F) = \sum_{\lambda\in\Lambda} F_{\langle\lambda\rangle} = (F)$. It proves the direct implication. The converse implication is immediate. □

4 Syzygies and Commutation Relations

In this section, we analyze more precisely the relations between the polynomials of the border basis $F = (f_\omega)_{\omega\in\partial B}$ where $f_\omega = \omega - \rho_\omega$ with $\rho_\omega \in \langle B\rangle$. These relations or syzygies form a module that we denote by

$$Syz(F) = \{\sum_\omega h_\omega e_\omega \in \mathbb{K}[\mathbf{x}]^{\partial B}; \sum_\omega h_\omega f_\omega = 0\},$$

where $(e_\omega)_{\omega\in\partial B}$ is the canonical basis of $\mathbb{K}[\mathbf{x}]^{\partial B}$. If F is a border basis family constructed from an initial set of polynomials $H = \{h_1, \ldots, h_s\}$, we can express $f_\omega \in F$ in terms of the polynomials in H (and conversely) so that a syzygy on F induces a syzygy on H (and conversely). Therefore, we are going here to consider only the syzygies on F.

For any $b \in \langle B\rangle$ and $i \in [1, n]$, $x_i\, b \in \langle B^+\rangle$ can be projected in $\langle B\rangle$ along F:

$$\pi_F(x_i\, b) = x_i\, b - \sum_{\omega\in\partial B} \mu^i_{b,\omega} f_\omega.$$

More generally, for any $p \in \mathbb{K}[\mathbf{x}]$, we denote by $\mu^i_{p,\omega}$ the coefficient $\mu^i_{b,\omega}$ in $\pi_F(x_i\, b)$ for $b = \pi_F(p) \in \langle B\rangle$ and by $\mu^i(p) = \sum_{\omega\in\partial B} \mu^i_p f_\omega$. Notice that if $x_i p \in B$ then $\mu^i(p) = 0$.

For any $m = x_{i_3} \cdots x_{i_k} \in B$ and $i_1, i_2 \in 1 \ldots n$, the two decompositions $\pi_F(x_{i_1} \pi_F(x_{i_2} m)) = \pi_F(x_{i_2} \pi_F(x_{i_1} m))$ yield the syzygy

$$x_{i_1} \sum_{\omega\in\partial B} \mu^{i_2}_{m,\omega} f_\omega - x_{i_2} \sum_{\omega\in\partial B} \mu^{i_1}_{m,\omega} f_\omega - \sum_{\omega\in\partial B} (\mu^{i_2}_{x_{i_1} m,\omega} - \mu^{i_1}_{x_{i_2} m,\omega})\, f_\omega = 0. \tag{1}$$

These relations can also be rewritten as:

$$x_{i_1}\mu^{i_2}(m) + \mu^{i_1}(x_{i_2}m) - x_{i_2}\mu^{i_1}(m) - \mu^{i_2}(x_{i_1}m) = 0.$$

We denote by Ξ the module of $\mathbb{K}[\mathbf{x}]^{\partial B}$ generated by these syzygies. It is also the module generated by the relations of commutation $M_{i_1} \circ M_{i_2}(b) - M_{i_2} \circ M_{i_1}(b) = 0$ for all $b \in B$, $i_i, i_2 \in [1 \ldots n]$. We distinguish the following relations:

- If $x_{i_1}m, x_{i_2}m \in B$, then $\mu^{i_1}(m) = \mu^{i_2}(m) = 0$, $\mu^{i_1}(x_{i_2}\, m) = \mu^{i_2}(x_{i-1}\, m) = 0$ and m does not yield a non-trivial relation of the form (1).
- If $x_{i_1}m \in B$ but $x_{i_2}m \in \partial B$, and $x_{i_1}\, x_{i_2}\, m \in \partial B$, then $\pi_F(x_{i_2}m) = x_{i_2}m - f_{x_{i_2}m}$ and we have the relation
$$x_{i_1}\, f_{x_{i_2}m} - f_{x_{i_1}x_{i_2}m} + \sum_{\omega\in\partial B} \mu^{i_1}_{x_{i_2}m,\omega} f_\omega = 0. \tag{2}$$
- If $x_{i_1}m \in B$, $x_{i_2}m \in \partial B$, and $x_{i_1}\, x_{i_2}\, m \in B$, then $\pi_F(x_{i_2}m) = x_{i_2}m - f_{x_{i_2}m}$ and we have the relation
$$x_{i_1}\, f_{x_{i_2}m} + \sum_{\omega\in\partial B} \mu^{i_1}_{x_{i_2}m,\omega} f_\omega = 0. \tag{3}$$
- If $x_{i_1}m \in \partial B$ and $x_{i_2}m \in \partial B$, then $\pi_F(x_{i_1}m) = x_{i_1}m - f_{x_{i_1}m}$, $\pi_F(x_{i_2}m) = x_{i_2}m - f_{x_{i_2}m}$ and we have the syzygy
$$x_{i_1} f_{x_{i_2}\, m} - x_{i_2} f_{x_{i_1}m} - \sum_{\omega\in\partial B} (\mu^{i_2}_{x_{i_1}m,\omega} - \mu^{i_1}_{x_{i_2}m,\omega}) f_\omega = 0 \tag{4}$$

The syzygies (2) and (4) are called respectively *next-door* and *across-the-street relations* in [16]. The syzygies (3) does not exist if B is stable under division (since in this case, $x_{i_2}m \notin B$ implies $x_{i_1}\, x_{i_2}m \notin B$). All these syzygies are simply the non-trivial relations induced by the "border" C-polynomials (see Definition 2.9).

Let us now prove that the module of syzygies is generated by these commutations syzygies. For any monomial $m = x_{i_1} \cdots x_{i_k}$, we can rewrite by induction its projection as:

$$\pi_F(m) = \pi_F(x_{i_1}\pi_F(x_{i_2}\cdots) = m - \sum_{l=1\ldots k} x_{i_1}\cdots x_{i_{l-1}} \sum_{\omega\in\partial B} \mu^{i_l}_{x_{i_{l+1}}\cdots x_k,\omega} f_\omega \tag{5}$$

with the convention that $x_{i_{k+1}} = 1$. We denote by

$$\Xi_{x_{i_1},\ldots,x_{i_k}} = \sum_{l=1\ldots k} x_{i_1}\cdots x_{i_{l-1}} \sum_{\omega\in\partial B} \mu^{i_l}_{x_{i_{l+1}}\cdots x_k,\omega} e_\omega$$

the corresponding element of $\mathbb{K}[\mathbf{x}]^{\partial B}$. Notice that this decomposition depends on the order of the decomposition $m = x_{i_1} \cdots x_{i_k}$ as a product of variables.

Lemma 4.1 *If* $m = x_{i_1} \cdots x_{i_k} = x_{j_1} \cdots x_{j_k}$ *then*

$$\Xi_{x_{i_1}\cdots x_{i_k}} - \Xi_{x_{j_1},\ldots,x_{j_k}} \in \Xi.$$

Proof. Consider first a permutation of two variables $m = m_1 x_{i_l} x_{i_{l+1}} m_2 = m_1 x_{i_{l+1}} x_{i_l} m_2$ (with $m_1 = x_{i_1} \cdots x_{i_{l-1}}$, $m_2 = x_{i_{l+2}} \cdots x_{i_k}$). Using (5), the two way of projecting $m = m_1 x_{i_l} x_{i_{l+1}} m_2 = m_1 x_{i_{l+1}} x_{i_l} m_2$ yield the syzygy

$$\begin{aligned}&\Xi_{\ldots,x_{i_l},x_{i_{l+1}},\ldots} - \Xi_{\ldots,x_{i_{l+1}},x_{i_l},\ldots} = m_1 \times \\ &\qquad\left(x_{i_l} \sum_{\omega\in\partial B} \mu^{i_{l+1}}_{m_2,\omega} e_\omega - x_{i_{l+1}} \sum_{\omega\in\partial B} \mu^{i_l}_{m_2,\omega} e_\omega + \sum_{\omega\in\partial B} (\mu^{i_l}_{x_{i_{l+1}} m_2,\omega} - \mu^{i_{l+1}}_{x_{i_l} m_2,\omega}) e_\omega \right)\end{aligned}$$

which is in Ξ. By iterated permutations of two variables, we transform the sequence $x_{i_1}, \ldots, x_{i_k}$ into the sequence $x_{j_1}, \ldots, x_{j_k}$. This allows us to rewrite $\Xi_{x_{i_1} \cdots x_{i_k}}$ into $\Xi_{x_{j_1},\ldots,x_{j_k}}$ modulo Ξ. □

Lemma 4.2 *If $m\,\theta = m'\,\theta'$ with $\theta, \theta' \in \partial B$ and $m, m' \in \mathcal{M}$, then*

$$m\, e_\theta \equiv m'\, e_{\theta'} + \sum_{\omega\in\partial B} p_\omega\, e_\omega \bmod \Xi$$

with $p_\omega \in \mathbb{K}[\mathbf{x}]$ of degree $< \max(|m|, |m'|)$.

Proof. If $m = x_{i_1} \cdots x_{i_d}$ with $\theta = x_{i_{d+1}} b \in \partial B$ and $b \in B$, the projection formula (5) of $m\,\theta$ has the form

$$\pi_F(m\,\theta) = m\,\theta - x_{i_1} \cdots x_{i_d} f_\theta - \sum_{l=1\ldots d} x_{i_1} \cdots x_{i_{l-1}} \sum_{\omega\in\partial B} \mu^{i_l}_{x_{i_{l+1}} \cdots x_{i_d} \theta,\omega} f_\omega$$

since $x_{i_{d+1}}\, b = \theta$ and $\pi_F(\theta) = \theta - f_\theta$. If $m' = x_{j_1} \cdots x_{j_{d'}}$ and $\theta' \in \partial B$ with $m\theta = m'\theta'$, by Lemma 4.1, the two decompositions yield the syzygy

$$\begin{aligned} & x_{i_1} \cdots x_{i_d} e_\theta - x_{j_1} \cdots x_{j_{d'}} e_{\theta'} \\ & + \sum_{l=1\ldots d} x_{i_1} \cdots x_{i_{l-1}} \sum_{\omega\in\partial B} \mu^{i_l}_{x_{i_{l+1}} \cdots x_{i_d} \theta,\omega} e_\omega \\ & - \sum_{l=1\ldots d'} x_{j_1} \cdots x_{j_{l-1}} \sum_{\omega\in\partial B} \mu^{j_l}_{x_{j_{l+1}} \cdots x_{j_{d'}} \theta',\omega} e_\omega \end{aligned}$$

as an element of Ξ. This syzygy is of the form $m\, e_\theta - m'\, e_{\theta'} + \sum_{\omega\in\partial B} p_\omega\, e_\omega$ with $\deg(p_\omega) < \max(d, d')$, which proves the lemma. □

This yields the following result for a border basis, conjectured in [16] for the case where B is stable by division:

Theorem 4.3 *Let $B \subset \mathcal{M}$ be connected to 1 and let $F = (f_\omega)_{\omega\in\partial B}$ be a border basis for B. Then $Syz(F)$ is generated by the relations (2), (3) and (4).*

Proof. Let $\sigma = \sum_\omega p_\omega\, e_\omega \in Syz(F)$. Consider in this sum, any term $\lambda\, m\, e_\theta$ where $\lambda \in \mathbb{K} - \{0\}$, $m \in \mathcal{M}$, $\theta \in \partial B$ and $\delta_B(m\,\theta) < |m| + 1$. Then, there exists $m' \in \mathcal{M}$, $\theta' \in \partial B$ such that $m\,\theta = m'\theta'$ and $\delta_B(m'\,\theta') = |m'| + 1$. This implies that $|m| > |m'|$. Applying lemma 4.2, the product $m\, e_\theta$ can be reduced modulo Ξ, to

$$m' e_{\theta'} + \sum_{\omega\in\partial B} q_\omega\, e_\omega,$$

with $\deg(q_\omega) < |m|$. Iterating this reduction, we may assume that each term $\lambda\, m\, e_\omega$ ($\lambda \in \mathbb{K} - \{0\}$, $m \in \mathrm{supp}(p_\omega)$, $\omega \in \partial B$) is such that $\delta_B(m\,\omega) = |m| + 1$. Notice that the polynomial $m\, f_\omega$ has only one monomial of maximal B-index, which is $m\,\omega$.

As $\sum_\omega p_\omega\, f_\omega = 0$, there exist $\theta \neq \theta' \in \partial B$ and monomials $m \in \mathrm{supp}(p_\theta)$, $m' \in \mathrm{supp}(p_{\theta'})$ such that $m\,\theta = m'\,\theta'$ and $\delta_B(m\,\theta)$ is maximal among all terms of the syzygy. By lemma 4.2, we can replace this pair $(m\, e_\theta, m'\, e_{\theta'})$ modulo Ξ by a sum of terms of smaller degree. This transformation reduces either the number of terms with maximal B-index or the degree of the polynomials p_ω.

Since we cannot iterate it infinitely, we deduce that σ is in Ξ. □

5 Algorithmic issues

From the preceding sections, we deduce an algorithm as it is done in [29]. The main idea is to translate the previous concepts into linear algebra. From Section 3 to compute effectively a normal form, one has to find a (monomial) basis B of the quotient algebra connected to 1, and border relations such that the multiplication operators commute. The algorithm described in [29], is a fix point method, which updates

- a potential monomial basis B of the quotient algebra,
- a set P of polynomials or rewriting rules (with one monomial of their support in ∂B and the remaining monomials in B),

until a fix point is reached. At each step of the algorithm, the following operations are performed:

1. The set P' of polynomials of P^+ with support in B^+ is computed.
2. By taking linear combinations, a basis $\tilde{P}$ of the vector space $\langle P' \rangle$ is computed, such that each element of this basis has at most one monomial in ∂B (and the other in B).
3. The C-polynomials of the elements of P with their support in B^+ are computed and reduced by $\tilde{P}$.
4. If non-zero polynomials with support in B appear, the potential basis B and the polynomial set P are updated. The update of B is done by removing some parts of B. The update of P is done by combining the elements of $\tilde{P}$ and the reduced C-polynomials, in order to get rewriting rules for the new set B.

The details and the technical proof of termination of the algorithm are available in [29]. We mention here that we have improved the algorithm described in this paper since then: the ways the degree drops are treated in [29] is now done to avoid to the repeated execution of similar reductions.

6 Stability of the bases

This section is devoted to the study of the stability under numerical perturbations, of the bases computed by the previous algorithm.

In many real-life problems, the system $\mathbf{f} = (f_1, \ldots, f_s)$ to be solved is given only with limited accuracy. However, most of the time, one also knows that the structure of the solutions is invariant in a small neighborhood of the system. Hence one of the feature that is often required to polynomial solvers is to produce a representation of the quotient algebra that is stable in a small neighborhood of the initial system. The structural numerical stability of the basis is expected in order to have a smooth behavior of the coefficients of the representation in the neighborhood of $\mathbf{f} = (f_1, \ldots, f_s)$. By definition, a neighborhood of $\mathbf{f}$ is an open set in the space of vector of polynomials $(h_1, \ldots, h_s)$ such that $\Lambda(h_i) \preceq \Lambda(f_i)$ $(i = 1, \ldots, n)$ and which contains $\mathbf{f}$. For $\epsilon > 0$, we define by $N_\epsilon(\mathbf{f})$ the set of systems $(h_1, \ldots, h_s)$ such that $\Lambda(h_i) \preceq \Lambda(f_i)$ and the coefficient vector of h_i is at most at distance ϵ from the coefficient vector of f_i (for the ∞-norm).

Assumption 6.1 *Hereafter, γ denotes a choice function refining a reducing grading Λ, such that for all $p \in R$, $\gamma(p)$ depends only on the support of p and not on the numerical value of its monomial coefficients (e.g. Macaulay's choice function, grevlex choice function,...). Let γ_ϵ be the choice function that for any $p \in R$, applies the choice function γ on the monomials of p, which coefficient norm is bigger than ϵ.*

Theorem 6.2 *Let $\mathbf{f} = (f_1, \ldots, f_s)$ be a zero dimensional polynomial system such that in a neighborhood U of $\mathbf{f}$, all systems have the same number D of complex solutions, counted with multiplicities. Then for all $\epsilon > 0$ small enough, there exists $\nu > 0$ such that for any system $\mathbf{f}' \in N_\nu(\mathbf{f}) \subset U$, the basis B computed with γ satisfying Assumption 6.1 for the system $\mathbf{f}$ is also a basis for the system $\mathbf{f}'$.*

Proof. Let us consider the matrix M whose rows correspond to the coefficient vectors of the monomial multiples $m\, f_i$, with $\deg(m) + \deg(f_i) \leq \kappa + d_0$ where κ is the number of loops in Algorithm [29] and d_0 the maximum degree of the polynomials f_i. The columns are indexed by all the monomials of degree $\leq \kappa + d_0$.

We denote by B the monomial set obtained as a basis of $\mathcal{A} = R/I$ by applying Algorithm [29] to $f_1, \ldots, f_s$ with the choice function γ. By construction, the set of monomials indexing the columns of M contains ∂B.

Let M^g be the same matrix as M but constructed with the polynomials f_i replaced by *generic* equations, i.e. equations with indeterminate coefficients having the same support as $f_1, \ldots, f_s$. Let N be the block of columns of M indexed by monomials not in B and let N^g be the corresponding block in M^g.

Since the operations of Algorithm [29] consist in computing linear combinations of some monomial multiples of the polynomials P (see Section 5 and [29]) and thus of monomial multiples of f_i of degree $\leq \kappa + d_0$, the complete computation can be reinterpreted as an optimized triangulation procedure of the block N.

The block N^g specialized at $\mathbf{f}$ is invertible, because any monomial not in B can be reduced by the computed border basis of $\mathbf{f}$ to an element in $\langle B \rangle$. This implies that N^g specialized at $\mathbf{f}'$ is also invertible, for $\mathbf{f}' \in N_\nu(\mathbf{f})$ with $\nu > 0$ sufficiently small. Therefore any monomial of ∂B can be reduced modulo the ideal $(\mathbf{f}')$ to an element in $\langle B \rangle$. Consequently, B (which contains 1) is a generating set of the quotient algebra $\mathcal{A}' = R/(\mathbf{f}')$ for any $\mathbf{f}' \in N_\nu(\mathbf{f})$.

As the number of solutions $D = |B|$ (counted with multiplicity) is left unchanged by small perturbations in the neighborhood $N_\nu(\mathbf{f})$ of $\mathbf{f}$, the monomial set B, that has exactly cardinality D, is also a basis of the quotient algebra $\mathcal{A}' = R/(\mathbf{f}')$ for the perturbed system $\mathbf{f}'$. □

Consider now a slight modification of Algorithm [29]: the coefficients whose norm is less than ϵ are simply ignored in all the steps of Algorithm [29]. This means that they will not be taken into account for choosing a *leading* monomial, or deciding if a polynomial is nonzero. This small variant will be denoted as the ϵ-algorithm in the next theorem. Remark here that this behavior is quite classical in fact, it is more or less what is usually done when neglecting small coefficients in a numerical algorithm. Remark also that the following theorem performs rigorously, and that the computed result is *not* an approximation of the true quotient algebra!

Then the following holds:

Theorem 6.3 *Let $\mathbf{f} = (f_1, \ldots, f_s)$ be a zero dimensional polynomial system such that in a neighborhood U of $\mathbf{f}$, all systems have the same number D of complex solutions, counted with multiplicities. Then for all $\epsilon > 0$ small enough, there exists $\nu > 0$ such that for any system $\mathbf{f}' \in N_\nu(\mathbf{f}) \subset U$, the basis B computed with γ satisfying Assumption 6.1 for the system $\mathbf{f}$ is also the basis obtained with γ_ϵ and the ϵ-algorithm for the system $\mathbf{f}'$.*

Proof. By Theorem 6.2, B is also a basis of the quotient algebra $R/(\mathbf{f}')$. This basis B is obtained by applying Algorithm [29] to $\mathbf{f}$. The result of this algorithm does not change for the system $\mathbf{f}$ if we replace the choice function γ by γ_ϵ for $\epsilon > 0$ small enough (eg. smaller than the minimum of the norm of the coefficients of the polynomials on which γ is applied).

Let us show by induction on the loop index k of the algorithm that the steps and the polynomials computed by the ϵ-algorithm with γ_ϵ are the same as for the direct algorithm, up to the terms of norm smaller that ϵ.

It is true obviously for the first step $k = 1$. Let us suppose now that steps $1, \ldots, k'$ of Algorithm [29] ran on $\mathbf{f}$ and its ϵ variant ran on $\mathbf{f}'$ are structurally the same and let us show that step $k'+1$ is also structurally the same for the two computations. The coefficients of all these constructed polynomials are rational functions of the coefficients of $\mathbf{f}'$, which are well defined in a neighborhood of $\mathbf{f}$. If $\mathbf{f}'$ is close enough to $\mathbf{f}$, the monomials which are in the support of the constructed polynomials for $\mathbf{f}'$ but not in the support of the constructed polynomials for $\mathbf{f}$, have coefficients of norm smaller then ϵ.

If, by hypothesis, the first k' steps are structurally identical, the same polynomials, up to terms of norm smaller than ϵ, appear when selecting the polynomials in P^+, and the same C-polynomials are constructed (see Section 5 and [29]). By continuity of the coefficients of the constructed polynomials, the *same* pivots (of norm bigger than ϵ) are used to construct the new elements in $\tilde{P}$. Similarly, in a neighborhood of $\mathbf{f}$, up to terms of small norm, the C-polynomials not reducing to zero are the same for the two computations. If choices of leading monomials are to be performed, then, by Definition 6.1, γ_ϵ will select the same monomials for $\mathbf{f}'$ and $\mathbf{f}$. Finally at the end of step $k'+1$ the same computations are performed, up to terms of small norm and the coefficients of the new constructed polynomials are rational functions of the coefficients of $\mathbf{f}'$, which are well defined in a neighborhood of $\mathbf{f}$.

This ends the induction showing that the two computations are structurally identical for $\mathbf{f}' \in N_\nu(\mathbf{f})$ with $\nu > 0$ small enough. Hence B will also be found as a basis of $R/(\mathbf{f}')$ with this ϵ-algorithm. □

The rewriting rules obtained from the ϵ-algorithm are close to the exact rewriting rules of the system $\mathbf{f}'$. Their numerical quality can be improved by iterative refinements such as Newton-like iterations using the commutation relations. Such approach has been investigated in [34].

Remark 6.4 (Numerical certification) *Theorem 6.3 shows the continuity of the normal form computation with respect to the coefficient of the input system. It states that there exists a region of* stability *for the computed quotient algebra representation, but it is an open problem to compute apriori the value ϵ and ν for a given polynomial system in order to control the size of the allowed perturbations. This problem is the subject of further work.*

Remark 6.5 (Flatness) *Theorem 6.3 also shows that if we consider a rationally parametrized family of systems $\mathbf{f}_t \in N_\nu(\mathbf{f})$ for all $t \in [0, 1]$, such that $\mathbf{f}_0 = \mathbf{f}$ and F_0 is the border basis for B, then the* *set B is also a basis of $\mathcal{A}_t = R/(\mathbf{f}_t)$. Moreover, the border basis F_t of $\mathbf{f}_t$ for B is of the form $F_t = (\omega - \rho_{\omega,t})_{\omega \in \partial B}$, where $\rho_{\omega,t} \in \langle B \rangle$ is a continuous (rational) function of t on $[0, 1]$ such that $F_0 = (\omega - \rho_{0,\omega})_{\omega \in \partial B}$. This also implies that the C-relations (2) and (4) generating the syzygies of F_t are continuous (rational) functions of $t \in [0, 1]$, which coincide with the C-polynomials relations of F_0 at $t = 0$. Consequently any syzygy of F_0 which is a combination of the C-relations can be deformed continuously into a syzygy of F_t (for $t \in [0, 1]$). In other words, the systems $\mathbf{f}'$ in the neighborhood $N_\nu(\mathbf{f})$ are* flat deformations *of the system* $\mathbf{f}$ *[7].*

7 Experimentation

The algorithm described in the previous section is implemented in the library SYNAPS[2]. It corresponds to about 50 000 lines of C++-code. It involves a direct sparse matrix solver. The numerical approximation of the roots are obtained by eigenvalues computation, using the library LAPACK (the routine `zgees`) and the strategy described in [5]. The computations are performed on an AMD-Athlon 2400+ with $256MB$ of main memory. We show the results obtained with our implementation in the case where the grading that we use for $\mathbb{K}[\mathbf{x}]$ is the usual one. In the sequel, *drvl* will refer to the choice function associated to the Degree Reverse Lexicographical order, *dlex* to the degree lexicographical order, Mac to Macaulay's choice function (see Example 2.8), *minsz* to the choice function over the rational that minimizes the memory needed in the reduction loop (this choice only minimizes a local step and does not insure local minimality of the global required memory), and *mix* to the choice function that returns randomly either the result of *minsz*, or of *drvl* applied to its input. To analyse the quality of approximation, we mesure the maximal norm at the computed roots of the initial polynomials f_i and denote it hereafter by *mnacr*.

7.1 Generic equations

The method we propose here is an extension of the Gröbner bases computations. As such it can compute Gröbner bases. The implementation we have is not as optimized as the Gröbner bases ones that are being worked on for decades. An important work, mostly on linear algebra, remains to be done on our program. However we want to show that the method we propose here is competitive, and that it does not lose the good practical efficiency of Gröbner bases computations [10]. As the arithmetic used in our programs for doing exact computation is the rational arithmetic of GMP, which is much slower than integer computations used in the other software we will restrict our-self to the use of modular arithmetic. The family of examples we have chosen is the $Katsura(n)$[3] equations. These equations are projective complete intersection with no zero at infinity. Using the Macaulay choice function, we know apriori that Macaulay's basis will be a monomial basis of the quotient algebra, so we know, apriori, what monomials will be leading monomials for the whole computation; so in this case we can guarantee that no test to 0 returns erroneous result even using floating point arithmetic. We compare first our program to one of the best implementations available, Magma's implementation of F_4 algorithm [10].

[2] `http://www-sop.inria.fr/galaad/synaps/`

[3] `http://www-sop.inria.fr/galaad/data/`

n	Synaps mac		Synaps drvl		Magma drvl	
7	0.19s	3M	0.22s	3M	0.05s	3M
9	6.17s	5M	8.44s	5M	1.670s	7M
10	32.39s	14M	56.84s	13M	13.50s	23M
11	252.05s	50M	387.97s	45M	96.76s	70M
12	1935.25s	191M	3072.08s	157M	1560.76s	240M

Let us mention that Gb, one of the reference implementation of Buchberger's algorithm, spends $659s$ on $Katsura(10)$.

Numerically we observe that choosing the *mac* function also results in a better conditioning of the computations. More precisely on *Katsura(6)*, and using a threshold of 10^{-10} we have:

γ	drvl	dlex	mac	drvl	dlex	mac	drvl	dlex	mac
# bits	128	128	128	80	80	80	64	64	64
time	1.98s	2.62s	1.64s	1.35s	3.98s	0.95s	–	–	0.9s
mnar	10^{-28}	10^{-24}	10^{-30}	10^{-20}	10^{-15}	10^{-19}	–	–	10^{-11}

For the 64 bits computation the results computed for the *drvl* and *dlex* orders are erroneous due to roundoff errors. The time given is the time spent in the computation of the multiplication matrices. Afterward, we used either LAPACK to perform the eigenvector computations or Maple when we needed extended precision. Because of the different nature of these tools, we do not report on the solving part timing. Finally we show here the amount of memory needed to perform the computations over $\mathbb{Q}$, using GMP `mpq`.

	mac	minsz	drvl	mix
time	4.22s	30.21s	6.54s	7.83s
size	4.2M	6.1M	4.4M	4.9M

On these experiments, we observe that the local strategy *minsz* which tends to minimize locally the size of the coefficients in the linear algebra operations, does not yield globally the optimal output size. In this example, the time and the memory size seem to be correlated.

7.2 Parallel robot

Let us consider the famous direct kinematic problem of the parallel robot[4] [23]. First we use floating point numbers to check the numerical requirements of the computations for different orders. For testing a number to be 0, we will use a leveling (here 10^{-8} is enough) and we will check afterward that the choices performed are the same as those done using modular arithmetic. This is equivalent to the use of an hybrid arithmetic [35].

γ	# bits	time	mnacr
drvl	128	2.07s	$0.3 * 10^{-24}$
dlex	128	4.27s	$0.3 * 10^{-23}$
mac	128	2.22s	$0.1 * 10^{-24}$

Here we see that choosing the right choice function can increase (but not so much in this case) the numerical accuracy of the roots. Hereafter we use the parametrization of [18] for solving, it involves more variables, gives better timings but less correct digits on the final result.

[4] `http://www-sop.inria.fr/galaad/data/`

# bits	time	mnacr
250	1.32s	10^{-63}
500	2.23s	10^{-140}

Finally we performed tests using rational arithmetic.

γ	mac	minsz	drvl	mix
time	315s	229.08s	201.65s	257.50
size	17M	14M	16M	13M

In fact, it is not so surprising to see that the choice function γ has a big impact in terms of the computational time and of the memory required. However in this problem, the time and the memory size do not seem to be correlated as in the previous case.

We also mention here that over-constraining the system can result in a dramatic decrease of the computation time. Indeed expressing more constraints than necessary can simplify computations significantly (see [29]).

Acknowledgments: We thank A. Quadrat for interesting discussions on differential algebra, prolongation and involutivity.

References

[1] W. Auzinger and H. J. Stetter. An elimination algorithm for the computation of all zeros of a system of multivariate polynomial equations. In *Proc. Intern. Conf. on Numerical Math.*, volume 86 of *Int. Series of Numerical Math*, pages 12–30. Birkhäuser Verlag, 1988.

[2] Y. Blinkov, V. Gerdt, and D. Yanovich. Construction of Janet bases (I. monomial bases, II. polynomial bases). *Computer Algebra in Scientific Computing / CASC 2001*, pages 233–263, 2001.

[3] L. Busé, M. Elkadi, and B. Mourrain. Using projection operators in computer aided geometric design. In *Topics in Algebraic Geometry and Geometric Modeling,*, pages 321–342. Contemporary Mathematics, 2003.

[4] E. Cartan. *Les systèmes différentiels extérieurs et leurs applications géométriques.* Paris, Hermann, 1945.

[5] R.M. Corless, P.M. Gianni, and B.M. Trager. A reordered Schur factorization method for zero-dimensional polynomial systems with multiple roots. In W.W. Küchlin, editor, *Proc. ISSAC*, pages 133–140, 1997.

[6] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra.* Undergraduate Texts in Mathematics. Springer Verlag, New York, 1992.

[7] D. Eisenbud. *Commutative Algebra with a view toward Algebraic Geometry*, volume 150 of *Graduate Texts in Math.* Berlin, Springer-Verlag, 1994.

[8] M. Elkadi and B. Mourrain. *Introduction à la résolution des systèmes d'équations algébriques*, volume 59 of *Mathématiques et Applications.* Springer-Verlag, 2007.

[9] I.Z. Emiris and B. Mourrain. Matrices in Elimination Theory. *J. of Symbolic Computation*, 28(1&2):3–44, 1999.

[10] J.C. Faugère. A new efficient algorithm for computing Gröbner Basis (F4). *J. of Pure and Applied Algebra*, 139:61–88, 1999.

[11] J.C. Faugère, P. Gianni, D. Lazard, and T. Mora. Efficient computation of zero-dimensional Gröbner bases by change of ordering. *J. of Symbolic Computation*, 16(4):329–344, 1993.

[12] M. Giusti, G. Lecerf, and B. Salvy. A Gröbner free alternative for polynomial system solving. *J. of Complexity*, 17(1):154–211, 2001.

[13] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.

[14] M. E. Huibregtse. An elementary construction of the multigraded Hilbert scheme of points. *Pacific Journal of Mathematics*, 223(2):269–315, 2006.

[15] A. Kehrein, M. Kreuzer, and L. Robbiano. An algebraist's view on border bases. In A. Dickenstein and I. Emiris, editors, *Solving Polynomial Equations: Foundations, Algorithms, and Applications.*, volume 14 of *Algorithms and Computation in Mathematics*, pages 169–202. Springer, 2005.

[16] A. Kehrein and Kreuzer. A characterisation of border bases. *J. Pure and Applied Algebra*, 196:251–270, 2005.

[17] M. Kuranishi. On E. Cartan's prolongation theorem of exterior differential systems. *American J. of Mathematics*, 79(1), 1957.

[18] D. Lazard. Stewart platforms and Gröbner bases. In *ARK'92*, Proceedings of Advance in Robot Kinematik, Ferrare, Italia, September 1992.

[19] F.S. Macaulay. Some formulae in elimination. *Proc. London Math. Soc.*, 1(33):3–27, 1902.

[20] F.S. Macaulay. *The Algebraic Theory of Modular Systems*. Cambridge Univ. Press, 1916.

[21] B. Malgrange. *Cartan involutivness = Mumford regularity*, volume 331 of *Comtemporary Mathematics*. 2003.

[22] H.M. Möller and T. Sauer. H-bases for polynomial interpolation and system solving. multivariate polynomial interpolation. *Advances Comput. Math.*, 12(4):335–362, 2000.

[23] B. Mourrain. The 40 generic positions of a parallel robot. In M. Bronstein, editor, *Proc. Intern. Symp. on Symbolic and Algebraic Computation*, ACM press, pages 173–182, Kiev (Ukraine), July 1993.

[24] B. Mourrain. Computing isolated polynomial roots by matrix methods. *J. of Symbolic Computation, Special Issue on Symbolic-Numeric Algebra for Polynomials*, 26(6):715–738, Dec. 1998.

[25] B. Mourrain. A new criterion for normal form algorithms. In M. Fossorier, H. Imai, Shu Lin, and A. Poli, editors, *Proc. AAECC*, volume 1719 of *LNCS*, pages 430–443. Springer, Berlin, 1999.

[26] B. Mourrain. *Pythagore's Dilemma, Symbolic-Numeric Computation and the Border Basis Method*, pages 223–243. Mathematics and Visualisation. Birkhäuser, 2006.

[27] B. Mourrain and Ph. Trébuchet. Solving projective complete intersection faster. In C. Traverso, editor, *Proc. Intern. Symp. on Symbolic and Algebraic Computation*, pages 231–238. New-York, ACM Press., 2000.

[28] B. Mourrain and Ph. Trébuchet. Algebraic methods for numerical solving. In *Proc. of the 3rd International Workshop on Symbolic and Numeric Algorithms for Scientific Computing'01 (Timisoara, Romania)*, pages 42–57, 2002.

[29] B. Mourrain and Ph. Trébuchet. Generalised normal forms and polynomial system solving. In M. Kauers, editor, *Proc. Intern. Symp. on Symbolic and Algebraic Computation*, pages 253–260. New-York, ACM Press., 2005.

[30] J.F. Pommaret. *Partial differential equations and group theory: new perspectives for applications.* Kluwer, 1994.

[31] G. Reid and Lihong Zhi. Solving nonlinear polynomial system via symbolic-numeric elimination method. In *Proc. of International Conference on Polynomial System Solving*, pages 50–53, 2004.

[32] F. Rouillier. Solving zero-dimensional polynomial systems through Rational Univariate Representation. *App. Alg. in Eng. Com. Comp.*, 9(5):433–461, 1999.

[33] D. J. Saunders. *The geometry of jet bundles.* Cambridge Univ. Press, Cambridge, 1989.

[34] H. J. Stetter. *Numerical polynomial algebra.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2004.

[35] C. Traverso and A. Zanoni. Numerical stability and stabilization of Gröbner basis computation. In T. Mora, editor, *Proc. Intern. Symp. on Symbolic and Algebraic Computation*, pages 262–269, New York, NY, USA, 2002. ACM Press.

[36] Ph. Trébuchet. *Vers une résolution stable et rapide des équations algébriques.* PhD thesis, Université Pierre et Marie Curie, 2002.